

Amendment to the Claims:

This listing of claims will replace all prior versions, and listings of claims in the application:

Listing of Claims:

1-5 (Canceled).

6. (Previously Presented) A computer-implemented method comprising:
- calculating, by a processor of the computer, a Levenshtein matrix of a first string and a second string;
 - determining a Levenshtein distance from said Levenshtein matrix;
 - determining a longest diagonal of equal hamming distance within the Levenshtein matrix;
 - determining a substring corresponding to the longest diagonal within said Levenshtein matrix, the substring being the largest common substring of the first and second strings;
 - calculating a Levenshtein score as a function of said Levenshtein distance;
 - calculating a largest common substring score as a function of said largest common substring;
 - storing at least one of: the Levenshtein matrix, the Levenshtein distance, the largest common substring, the Levenshtein score, and the largest common substring score in a computer-readable storage medium;
 - determining a similarity between said first string and said second string as a function of said Levenshtein score and said largest common substring score; and
 - based upon said similarity, automating at least one of: data entry for a database, processing within a database, or reporting from a database, the database including at least one of said first or second strings.
7. (Previously Presented) The method according to Claim 6, further comprising calculating an acronym score of said first string and said second string.

8. (Original) The method according to Claim 7, further comprising calculating a weighted acronym score comprising a product of said acronym score and an acronym weight factor.
9. (Original) The method according to Claim 6, further comprising:
 - calculating a weighted Levenshtein score comprising a product of said Levenshtein score and a Levenshtein weight factor;
 - calculating a weighted largest common substring score comprising a product of said largest common substring score and a largest common substring weight factor; and
 - calculating a Levenshtein/largest common substring score comprising a sum of said weighted Levenshtein score and said weighted largest common substring score.
10. (Original) The method according to Claim 9, wherein a sum of said Levenshtein weight factor and said largest common substring weight factor is equal to one.
11. (Original) The method according to Claim 9, further comprising calculating a first weighted numerical score comprising a product of said Levenshtein/largest common substring score and a string weight factor.
12. (Previously Presented) The method according to Claim 11, further comprising:
 - calculating an acronym score of said first string and said second string;
 - calculating a weighted acronym score comprising a product of said acronym score and an acronym weight factor; and
 - calculating a second weighted numerical score comprising a sum of said first weighted numerical score and said weighted acronym score.
13. (Original) The method according to Claim 12, wherein a sum of said string weight factor and said acronym weight factor is equal to one.

14. (Currently Amended) A computer-readable storage medium containing one or more sequences of executable instructions which cause a computing device to implement a method for determining a similarity comprising:

- calculating a Levenshtein matrix of a first string and a second string;
- determining a Levenshtein distance from said Levenshtein matrix;
- determining a longest diagonal of equal hamming distance within the Levenshtein matrix;
- determining a substring corresponding to the longest diagonal within said Levenshtein matrix, the substring being the largest common substring of the first and second strings;
- calculating a Levenshtein score as a function of said Levenshtein distance;
- calculating a largest common substring score as a function of said largest common substring;
- storing at least one of: the Levenshtein matrix, the Levenshtein distance, the largest common substring, the Levenshtein score, and the largest common substring score in a computer-readable storage medium;
- calculating a first numerical score as a function of said Levenshtein score and said largest common substring score; and
- based upon said ~~similarity~~ first numerical score, automating at least one of: data entry for a database, processing within a database, or reporting from a database, the database including at least one of said first or second strings.

15. (Previously Presented) The computer-readable medium according to Claim 14, wherein calculating said Levenshtein score comprises:

- subtracting the resultant of dividing said Levenshtein distance by an average of a length of said first string and a length of said second string from one.

16. (Previously Presented) The computer-readable medium according to Claim 14, wherein calculating said largest common substring score comprises:

- determining a length of said largest common substring from said Levenshtein matrix; and

dividing said length of said largest common substring by an average of a length of said first string and a length of said second string.

17. (Original) The computer-readable medium according to Claim 14, wherein calculating said first numerical score comprises:

- calculating a weighted Levenshtein score comprising a product of said Levenshtein score and a Levenshtein weight factor;

- calculating a weighted largest common substring score comprising a product of said largest common substring score and a largest common substring weight factor; and

- summing said weighted Levenshtein score and said weighted largest common substring score.

18. (Previously Presented) The computer-readable medium according to Claim 14, further comprising:

- calculating an acronym score;

- calculating a second numerical score as a function of said first numerical score and said acronym score; and

- further automating at least one of said data entry, processing or reporting based upon said second numerical score.

19. (Original) The computer-readable medium according to Claim 18, wherein calculating said second numerical score comprises:

- calculating a weighted Levenshtein score comprising a product of said Levenshtein score and a Levenshtein weight factor;

- calculating a weighted largest common substring score comprising a product of said largest common substring score and a largest common substring weight factor;

- calculating a Levenshtein/largest common substring score comprising a sum of said weighted Levenshtein score and said weighted largest common substring score;

calculating a weighted Levenshtein/largest common substring score comprising a product of said Levenshtein/largest common substring score and a Levenshtein/largest common substring weight factor;

calculating a weighted acronym score comprising a product of said acronym score and an acronym score weight factor; and

summing said weighted Levenshtein/largest common substring score and said weighted acronym score.

20. (Previously Presented) The computer-readable medium according to Claim 19, further comprising:

utilizing said first numerical score for automating said at least one of data entry, processing or reporting, when said first string and said second string comprise numerical-type strings; and

utilizing said second numerical score for automating said at least one of data entry, processing or reporting, when said first string or said second string comprise character-type strings.

21. (New) A computer-readable storage medium containing one or more sequences of executable instructions which cause a computing device to implement a method for determining a similarity, the method comprising:

calculating a Levenshtein matrix of a first string and a second string;
determining a Levenshtein distance from said Levenshtein matrix;
determining a longest diagonal of equal hamming distance within the Levenshtein matrix;
determining a substring corresponding to the longest diagonal within said Levenshtein matrix, the substring being the largest common substring of the first and second strings;
calculating a Levenshtein score as a function of said Levenshtein distance;
calculating a largest common substring score as a function of said largest common substring;

storing at least one of: the Levenshtein matrix, the Levenshtein distance, the largest common substring, the Levenshtein score, and the largest common substring score in a computer-readable storage medium;

determining a similarity between said first string and said second string as a function of said Levenshtein score and said largest common substring score; and

based upon said similarity, automating at least one of: data entry for a database, processing within a database, or reporting from a database, the database including at least one of said first or second strings.

22. (New) The computer-readable storage medium according to Claim 21, said method further comprising calculating an acronym score of said first string and said second string.

23. (New) The computer-readable storage medium according to Claim 22, said method further comprising calculating a weighted acronym score comprising a product of said acronym score and an acronym weight factor.

24. (New) The computer-readable storage medium according to Claim 21, said method further comprising:

calculating a weighted Levenshtein score comprising a product of said Levenshtein score and a Levenshtein weight factor;

calculating a weighted largest common substring score comprising a product of said largest common substring score and a largest common substring weight factor; and

calculating a Levenshtein/largest common substring score comprising a sum of said weighted Levenshtein score and said weighted largest common substring score.

25. (New) The computer-readable storage medium according to Claim 24, wherein a sum of said Levenshtein weight factor and said largest common substring weight factor is equal to one.

26. (New) The computer-readable storage medium according to Claim 24, said method further comprising calculating a first weighted numerical score comprising a product of said Levenstein/largest common substring score and a string weight factor.

27. (New) The computer-readable storage medium according to Claim 26, said method further comprising:

- calculating an acronym score of said first string and said second string;
- calculating a weighted acronym score comprising a product of said acronym score and an acronym weight factor; and
- calculating a second weighted numerical score comprising a sum of said first weighted numerical score and said weighted acronym score.

28. (New) The computer-readable storage medium according to Claim 27, wherein a sum of said string weight factor and said acronym weight factor is equal to one.

29. (New) An apparatus comprising:

- a processor; and
- a computer-readable storage medium in communication with the processor, the storage medium storing a sequence of executable instructions which cause the processor to perform a method comprising:
 - calculating a Levenshtein matrix of a first string and a second string;
 - determining a Levenshtein distance from said Levenshtein matrix;
 - determining a longest diagonal of equal hamming distance within the Levenshtein matrix;
 - determining a substring corresponding to the longest diagonal within said Levenshtein matrix, the substring being the largest common substring of the first and second strings;
 - calculating a Levenshtein score as a function of said Levenshtein distance;
 - calculating a largest common substring score as a function of said largest common substring;

storing at least one of: the Levenshtein matrix, the Levenshtein distance, the largest common substring, the Levenshtein score, and the largest common substring score in the computer-readable storage medium;

determining a similarity between said first string and said second string as a function of said Levenshtein score and said largest common substring score; and

based upon said similarity, automating at least one of: data entry for a database, processing within a database, or reporting from a database, the database including at least one of said first or second strings.

30. (New) The apparatus according to Claim 29, said method further comprising calculating an acronym score of said first string and said second string.

31. (New) The apparatus according to Claim 30, said method further comprising calculating a weighted acronym score comprising a product of said acronym score and an acronym weight factor.

32. (New) The apparatus according to Claim 29, said method further comprising:
calculating a weighted Levenshtein score comprising a product of said Levenshtein score and a Levenshtein weight factor;
calculating a weighted largest common substring score comprising a product of said largest common substring score and a largest common substring weight factor; and
calculating a Levenshtein/largest common substring score comprising a sum of said weighted Levenshtein score and said weighted largest common substring score.

33. (New) The apparatus according to Claim 32, wherein a sum of said Levenshtein weight factor and said largest common substring weight factor is equal to one.

34. (New) The method according to Claim 32, said method further comprising calculating a first weighted numerical score comprising a product of said Levenstein/largest common substring score and a string weight factor.

35. (New) The apparatus according to Claim 34, said method further comprising:
calculating an acronym score of said first string and said second string;
calculating a weighted acronym score comprising a product of said acronym score and an acronym weight factor; and
calculating a second weighted numerical score comprising a sum of said first weighted numerical score and said weighted acronym score.

36. (New) The apparatus according to Claim 35, wherein a sum of said string weight factor and said acronym weight factor is equal to one.

37. (New) A computer-implemented method comprising:
calculating, by a processor of the computer, a Levenshtein matrix of a first string and a second string;
determining a Levenshtein distance from said Levenshtein matrix;
determining a longest diagonal of equal hamming distance within the Levenshtein matrix;
determining a substring corresponding to the longest diagonal within said Levenshtein matrix, the substring being the largest common substring of the first and second strings;
calculating a Levenshtein score as a function of said Levenshtein distance;
calculating a largest common substring score as a function of said largest common substring;
storing at least one of: the Levenshtein matrix, the Levenshtein distance, the largest common substring, the Levenshtein score, and the largest common substring score in a computer-readable storage medium;
calculating a first numerical score as a function of said Levenshtein score and said largest common substring score; and

based upon said first numerical score, automating at least one of: data entry for a database, processing within a database, or reporting from a database, the database including at least one of said first or second strings.

38. (New) The method according to Claim 37, wherein calculating said Levenshtein score comprises:

subtracting the resultant of dividing said Levenshtein distance by an average of a length of said first string and a length of said second string from one.

39. (New) The method according to Claim 37, wherein calculating said largest common substring score comprises:

determining a length of said largest common substring from said Levenshtein matrix; and
dividing said length of said largest common substring by an average of a length of said first string and a length of said second string.

40. (New) The method according to Claim 37, wherein calculating said first numerical score comprises:

calculating a weighted Levenshtein score comprising a product of said Levenshtein score and a Levenshtein weight factor;

calculating a weighted largest common substring score comprising a product of said largest common substring score and a largest common substring weight factor; and

summing said weighted Levenshtein score and said weighted largest common substring score.

41. (New) The method according to Claim 37, further comprising:

calculating an acronym score;

calculating a second numerical score as a function of said first numerical score and said acronym score; and

further automating at least one of said data entry, processing or reporting based upon said second numerical score.

42. (New) The method according to Claim 41, wherein calculating said second numerical score comprises:

- calculating a weighted Levenshtein score comprising a product of said Levenshtein score and a Levenshtein weight factor;

- calculating a weighted largest common substring score comprising a product of said largest common substring score and a largest common substring weight factor;

- calculating a Levenshtein/largest common substring score comprising a sum of said weighted Levenshtein score and said weighted largest common substring score;

- calculating a weighted Levenshtein/largest common substring score comprising a product of said Levenshtein/largest common substring score and a Levenshtein/largest common substring weight factor;

- calculating a weighted acronym score comprising a product of said acronym score and an acronym score weight factor; and

- summing said weighted Levenshtein/largest common substring score and said weighted acronym score.

43. (New) The method according to Claim 42, further comprising:

- utilizing said first numerical score for automating said at least one of data entry, processing or reporting, when said first string and said second string comprise numerical-type strings; and

- utilizing said second numerical score for automating said at least one of data entry, processing or reporting, when said first string or said second string comprise character-type strings.

44. (New) An apparatus comprising:

- a processor; and

a computer-readable storage medium in communication with the processor, the storage medium storing a sequence of executable instructions which cause the processor to perform a method comprising:

- calculating a Levenshtein matrix of a first string and a second string;
- determining a Levenshtein distance from said Levenshtein matrix;
- determining a longest diagonal of equal hamming distance within the Levenshtein matrix;
- determining a substring corresponding to the longest diagonal within said Levenshtein matrix, the substring being the largest common substring of the first and second strings;
- calculating a Levenshtein score as a function of said Levenshtein distance;
- calculating a largest common substring score as a function of said largest common substring;
- storing at least one of: the Levenshtein matrix, the Levenshtein distance, the largest common substring, the Levenshtein score, and the largest common substring score in the computer-readable storage medium;
- calculating a first numerical score as a function of said Levenshtein score and said largest common substring score; and
- based upon said first numerical score, automating at least one of: data entry for a database, processing within a database, or reporting from a database, the database including at least one of said first or second strings.

45. (New) The apparatus according to Claim 44, wherein calculating said Levenshtein score comprises:

- subtracting the resultant of dividing said Levenshtein distance by an average of a length of said first string and a length of said second string from one.

46. (New) The apparatus according to Claim 44, wherein calculating said largest common substring score comprises:

- determining a length of said largest common substring from said Levenshtein matrix; and

dividing said length of said largest common substring by an average of a length of said first string and a length of said second string.

47. (New) The apparatus according to Claim 44, wherein calculating said first numerical score comprises:

calculating a weighted Levenshtein score comprising a product of said Levenshtein score and a Levenshtein weight factor;

calculating a weighted largest common substring score comprising a product of said largest common substring score and a largest common substring weight factor; and

summing said weighted Levenshtein score and said weighted largest common substring score.

48. (New) The apparatus according to Claim 44, said method further comprising:

calculating an acronym score;

calculating a second numerical score as a function of said first numerical score and said acronym score; and

further automating at least one of said data entry, processing or reporting based upon said second numerical score.

49. (New) The apparatus according to Claim 48, wherein calculating said second numerical score comprises:

calculating a weighted Levenshtein score comprising a product of said Levenshtein score and a Levenshtein weight factor;

calculating a weighted largest common substring score comprising a product of said largest common substring score and a largest common substring weight factor;

calculating a Levenshtein/largest common substring score comprising a sum of said weighted Levenshtein score and said weighted largest common substring score;

calculating a weighted Levenshtein/largest common substring score comprising a product of said Levenshtein/largest common substring score and a Levenshtein/largest common substring weight factor;

calculating a weighted acronym score comprising a product of said acronym score and an acronym score weight factor; and

summing said weighted Levenshtein/largest common substring score and said weighted acronym score.

50. (New) The apparatus according to Claim 19, said method further comprising:

utilizing said first numerical score for automating said at least one of data entry, processing or reporting, when said first string and said second string comprise numerical-type strings; and

utilizing said second numerical score for automating said at least one of data entry, processing or reporting, when said first string or said second string comprise character-type strings.